

KERNIHAN B W, RITCHIE D M. C 程序设计语言[M]. 徐宝文, 李志, 译. 2 版. 北京: 机械工业出版社, 2004: 27-28.

(赵启阳 金靖)

### 1.2.1.3 头文件与名字空间的概念

头文件是后缀名为“.h”的文件, 通过文件包含指令(即#include 指令), 可以在编译期间将指定头文件的内容包含进当前的文件中。不同的头文件包含不同的功能, 可以提供给程序设计者使用, 节省程序设计人员的重复劳动。程序设计者只需用一行#include 命令就可以使用这些功能, 大大地提高了编程效率。

在源文件中, 任何形如#include "filename"或#include <filename>的行都将被替换为 filename 指定的文件的内容。如果 filename 用双引号括起来, 则在源文件所在的位置查找该文件。如果在该位置没有找到文件, 或者 filename 是用<>括起来的, 则在系统目录中查找该文件。如果找不到文件, 编译系统将显示出错信息。被包含的文件本身也可以包含#include 指令。

C++语言支持 ANSI 标准定义的函数库(简称标准库), 标准库中的函数、类型和宏在标准头文件中定义。在 C++语言中可以包含标准模板库(standard template library), 使程序设计者更方便地编写程序。

在算法竞赛中, 使用"bits/stdc++.h"较为流行, 这个头文件中包含了几乎所有 C++库中的头文件, 因此一般只需要包含这一个头文件就能满足所有的需求, 简单便捷。但从另一个角度来看, 由于包含了所有头文件, 因此编译的速度自然就稍慢一些, 增加了编译时间。而且这不是 C++的标准库, 因而不是所有的编译系统都有。

C++语言是在 C 语言的基础上开发的, 早期的 C++语言编译器是将 C++程序转换为 C 程序, 再通过 C 编译器完成编译。当时 C++使用 C 语言的函数库, C++头文件仍然以“.h”为后缀, 它们所包含的类、函数、宏等都是全局范围的。但如果有多人开发软件, 由于各自分工, 可能会出现变量或函数命名冲突的问题。为了解决这类问题, C++引入了命名空间(namespace)的概念。

namespace 是 C++中的关键字, 用来定义一个命名空间。命名空间由{}包围, 可包含变量、函数、类等。“::”称为域解析操作符, 在 C++中用来指明要使用的命名空间。除了直接使用域解析操作符, 还可以采用 using 声明。

自 C++引入了命名空间的概念后, 为了保持兼容性, C++语言保留原来的头文件, 保证其在 C++中可以继续使用, 然后再把原来的函数库复制一份, 并在此基础上稍加修改, 把变量、函数、类等纳入命名空间 std(std 意为 standard)下, 即“标准命名空间”。这两份相似的函数库就可以分别被不同时期开发的程序同时使用, 不会产生冲突。

为了避免头文件重名, 原来 C 语言的头文件的名字前添加了字母 c, 比如 stdio.h 变成了 cstdio.h。C++函数库也对头文件的命名做了调整, 去掉了后缀“.h”, 比如

iostream.h 变成了 iostream。一般来说，对于不带“.h”的头文件，其变量、函数都位于命名空间 std 中，使用时需要声明命名空间 std。而对于带“.h”的头文件，则没有使用任何命名空间，所有符号都位于全局作用域中。

## 延伸阅读

[1] KERNIHAN B W, RITCHIE D M. C 程序设计语言[M]. 徐宝文, 李志, 译. 2 版. 北京: 机械工业出版社, 2004: 69-70.

[2] BJARNE S. C++程序设计语言(第1~3部分)[M]. 王刚, 杨巨峰, 译. 4 版. 北京: 机械工业出版社, 2016: 74-76, 337-359.

(金靖)

### 1.2.1.4 编辑、编译、解释、调试的概念

C++是高级语言，对 C++源程序文件(即.cpp)进行编写，称为“编辑”源程序。

高级语言需要被翻译为机器语言才能被计算机执行。

编译型语言：使用编译器程序，把文本形式的源代码翻译成机器语言，并形成目标文件。C++是一种编译型语言。

解释型语言：程序不需要编译，而是由语言特定的解释程序执行。Python 是典型的解释型语言。

调试：运行编写的程序，跟踪程序运行，发现并修正程序错误的过程。

## 延伸阅读

BJARNE S. C++程序设计语言(第1~3部分)[M]. 王刚, 杨巨峰, 译. 4 版. 北京: 机械工业出版社, 2016: 362-384.

(金靖)

## 1.2.2 基本数据类型

### 1.2.2.1 整数型：int、long long

不同数据类型的变量占用的存储空间大小不同。以下是两种常用的整数类型。

(1) int，整数类型，简称整型，占用 4 个字节，取值范围是 $-2^{31} \sim 2^{31}-1$ ，大约能够表示绝对值不超过  $2.1 \times 10^9$  的整数。

(2) long long，64 位整数类型，简称长整型，占用 8 个字节，又称 int 64，即用 64 个 bit 来存储，取值范围是 $-2^{63} \sim 2^{63}-1$ ，大约能够表示绝对值不超过  $9.2 \times 10^{18}$  的整数。

## 延伸阅读

[1] KERNIHAN B W, RITCHIE D M. C 程序设计语言[M]. 徐宝文, 李志, 译. 2 版. 北京: 机械工业出版社, 2004: 27-28.

[2] BJARNE S. C++程序设计语言(第1~3部分)[M]. 王刚, 杨巨峰, 译. 4版. 北京: 机械工业出版社, 2016: 124-126.

(金靖)

### 1.2.2.2 实数型: float、double

实数型的存储方式, 是由一个整数或定点数(即尾数)乘以某个基数(计算机中通常是2)的整数次幂。这种表示方法类似基数为10的科学计数法, 这种设计可以在某个固定长度的存储空间内表示更大范围的数。常用的实数型包括 float 和 double。

(1) float, 单精度实数型, 取值范围是 $-3.4 \times 10^{38} \sim 3.4 \times 10^{38}$ , 占用4个字节, 精度约为7位有效数字。

(2) double, 双精度实数型, 取值范围是 $-1.7 \times 10^{308} \sim 1.7 \times 10^{308}$ , 占用8个字节, 精度约为15位有效数字。

### 延伸阅读

[1] KERNIHAN B W, RITCHIE D M. C 程序设计语言[M]. 徐宝文, 李志, 译. 2版. 北京: 机械工业出版社, 2004: 27-28.

[2] BJARNE S. C++程序设计语言(第1~3部分)[M]. 王刚, 杨巨峰, 译. 4版. 北京: 机械工业出版社, 2016: 126-127.

(金靖)

### 1.2.2.3 字符型: char

char, 字符型, 占用1个字节, 表示一个字符, 如'a', '0' (必须使用英文单引号括起来)。char 类型中存放字符的是 ASCII 码, 因此字符型变量可以转化为整数, 取值范围为-128~127。

### 延伸阅读

[1] KERNIHAN B W, RITCHIE D M. C 程序设计语言[M]. 徐宝文, 李志, 译. 2版. 北京: 机械工业出版社, 2004: 27-28.

[2] BJARNE S. C++程序设计语言(第1~3部分)[M]. 王刚, 杨巨峰, 译. 4版. 北京: 机械工业出版社, 2016: 121-124.

(金靖)

### 1.2.2.4 布尔型: bool

bool, 布尔型, 用于表示真、假逻辑值, 占用1个字节。可以用整数对 bool 类型变量赋值, 赋0值即为 false, 非0值即为 true。

### 延伸阅读

BJARNE S. C++程序设计语言(第1~3部分)[M]. 王刚, 杨巨峰, 译. 4版. 北京: 机械工业出版社, 2016: 119-120.

(金靖)

## 1.2.3 程序基本语句

### 1.2.3.1 cin 语句、scanf 语句、cout 语句、printf 语句、赋值语句、复合语句

#### 1. cin 语句

cin(character input), 在 iostream 头文件中定义, 可以控制来自标准输入流 stdin 的输入。

```
int a;  
std::cin >> a;
```

#### 2. scanf 语句

scanf 是格式化输入函数, 在 cstdio 头文件中定义, 从 stdin 中读取数据。

scanf 的语法为:

```
scanf(格式字符串, 变量列表)
```

其返回值为成功赋值的参数的数量, 可以为 0; 若在赋值首个接收的参数前输入失败则为文件末尾(end of file, EOF)。

scanf 函数的格式字符串由非空白多字节字符、空白符和转换说明符%三类符号组成。转换说明符由以下几类组成:

- 1) %c, 代表一个字符;
- 2) %s, 代表一个字符串, 会读入一个到空白字符或者 EOF 为止的字符串;
- 3) %d, 代表一个十进制整数;
- 4) %u, 代表一个无符号十进制整数;
- 5) %f, 代表一个十进制的实数。

```
int a;  
long long b;  
char s[10];  
scanf("%d%lld%s", &a, &b, s);
```

#### 3. cout 语句

cout(character output), 在 iostream 头文件中定义, 可以控制来自标准输出流 stdout 的输出。可以用 fixed 和 setprecision() 来保留指定位数的浮点数。

```
#include <iostream>  
#include <iomanip> //控制精度需要包含  
using namespace std;  
int main() {  
    double d = 1234.5678912;
```

```

cout << 123 << " NOI" << endl;
cout << d << endl; // 默认保留 6 位有效数字,输出为 1234.57
cout << setprecision(3) << d << endl; // 保留 3 位有效数字,输出为 1.23e+003
cout << setprecision(9) << d << endl; // 保留 9 位有效数字,输出为 1234.56789
cout << setiosflags(ios::fixed); // 设置小数部分的精度
cout << d << endl; // 小数部分保留 9 位有效数字,输出为 1234.567891200
cout << fixed << setprecision(3) << d << endl; // 小数部分保留 3 位有效数字,输出为 1234.568

return 0;
}

```

#### 4. printf 语句

printf 是格式化输出函数，在 `cstdio` 头文件中定义，可以输出到 `stdout`。

printf 的语法为：

```
printf(格式字符串,表达式列表)
```

其格式字符串由普通多字节字符和转换说明符%构成。转换说明符类似 `scanf` 函数中的定义。

```

#include <cstdio>
#include <cmath>
using namespace std;
int main() {
    int a = 1;
    double pi = std::acos(-1.0);
    printf("a = %d, pi = %.9lf\n", a, pi); //输出 a = 1, pi = 3.141592654
    return 0;
}

```

#### 5. 赋值语句

形如  $a=b$  的语句是基本赋值语句，作用是将  $b$  的值赋值给  $a$ ， $b$  可以是常量、变量或表达式。

#### 6. 复合语句

复合语句由一对大括号和中间的语句组成，在语法上等价于单条语句。

### 延伸阅读

- [1] KERNIHAN B W, RITCHIE D M. C 程序设计语言[M]. 徐宝文, 李志, 译. 2 版. 北京: 机械工业出版社, 2004: 133-138.
- [2] BJARNE S. C++程序设计语言(第 1~3 部分)[M]. 王刚, 杨巨峰, 译. 4 版. 北京: 机械工业出版社, 2016: 78-80.